

Evaluation Scoring

Remember to be consistent with scoring methodology

0	Did not provide an answer
2	Incoherent, communicates a lack of understanding and/or capability
4	Marginal, communicates a weak knowledge, understanding, and/or capability
6	Average, communicates an average knowledge, understanding, and/or capability
8	Good, communicates that they are good at what they do and have a good understanding
10	Excellent, communicates that they are not just good, but provide value added services and capabilities that may exceed expectations

Give one score for each question - the average of all scores will be the total points awarded for the interview.

Vendor: Emagined		K689 - Security Assessment Services					
		Interview Total 10 pts					
Question 1	Many of the state agencies and local governments we work with have thick client applications that we test. Can you describe your experience with penetration testing thick client applications? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a thick client?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Question 2	Our engagements can include testing of SCADA systems. In some cases this testing must be done in production on critical systems such as water or sewer treatment plants. Can you describe your penetration testing experience with SCADA systems? What are some of the key questions you would have for the client to ensure a smooth valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Question 3	Our engagements sometimes include testing mainframes. These would typically be legacy applications that support core or enterprise functions, and therefore may not be as stable or robust as more modern applications. Can you describe your experience with penetration testing mainframes? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a mainframe?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Question 4	Our engagements sometimes include mobile applications? Can you describe your experience with penetration testing mobile systems? What are some of the key questions you would have for the client to ensure a smooth and valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Question 5	Can you describe a time when something went wrong during a prior penetration testing engagement? How did you find out something went wrong? How did you respond or correct the situation? What lessons were learned from that event?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					9	10	
Question 6	From time to time we need to have more than one penetration testing engagement going simultaneously- with each engagement likely having internal and external testing with approximately 4-6 applications and network testing. Would your firm be able to staff this? If so, how much lead time would your firm require? How many penetration staff would you be able to commit to each engagement? How long would it take your staff to complete a typical penetration test of this size?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Question 7	Communication is an important part of penetration testing work, especially because the systems we test are not managed by SAO, but rather our auditees. Can you tell us about any experience you have testing a third-party application, for example a vendor for your direct client? What type of communication challenges did you face? How did you overcome those challenges?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
						9	10
Average Score	9.71						

We have tested hundreds of different types of thick client applications, everything from custom apps that run in Citrix and other hardened environments, to clients that connect to web services, use APIs, or connect to databases. Our penetration testers have experience with not only testing the application but also with coding, so when we want to look at where that application is communicating, we have that expertise. We have also identified several zero day vulnerabilities on thick client applications that use CICS data. After bringing these vulnerabilities up to the customer, we were also able to meet with the software vendor and show them the vulnerabilities, so that they could develop and implement a remediation.

With thick client applications, we want to know if there's a licensing restriction; want to know if we can load it onto one of our systems or if it needs to be on one of the customer's systems; any kind of domain authentication or SSO requirement; want to have credentials to test the application; want to know what kinds of roles the application has; want to know what the application does; want to know if we're connecting to production data or non-production data; and if there are any binaries we can look at as part of testing.

We have done testing in SCADA environments for over 20 years. In the last 5 years, we've done at least a dozen water or sewer treatment plants; most of this testing has been in production, and some of it was on older and more sensitive gear (PLCs). Also have experience with other SCADA systems, including heating and cooling systems.

One of the first questions we would ask is if it is an air-gapped network, or if there's any convergence between the OT and IT networks (can someone get from one network to the other). Same question regarding internet connectivity. Once we get into the SCADA testing itself, we look at the various types of assets (PLCs, modbus systems, etc.). We've done traffic control systems as well. I don't believe there is a sector we haven't tested just yet.

We also do pentesting for some of the world's largest manufacturers of building supplies. This includes testing much older systems that are used to cut product to specification. The tolerance for downtime on these systems is very low because downtime results in production delays. Also have experience testing SCADA systems in paper mills, which are very sensitive environments.

Also helpful to gauge the level of operator knowledge in the customer environment, both in terms of IT knowledge and OT knowledge.

We have seen mainframes quite a bit over the 22 years that we've been doing penetration testing. This includes work with credit card manufacturers that use mainframes in their environments, as well as local and city governments that have relied on mainframe systems that date back to the 1990s. We treat mainframes very similar to how we treat SCADA networks; since those mainframes have ports and protocols that might be more fragile, we do more manual penetration testing on those mainframes. We have scripts and processes that we've created specifically for mainframe systems. We've generally tested 10-15 mainframe systems per year, each which often contains multiple systems.

Before conducting a mainframe test, need to know any type of connectivity that those mainframes use; whether or not they have any custom applications, APIs, or middleware that allow them to communicate with web applications or thick clients; need to know what types of roles exist on the mainframe and whether we could test with those accounts; what type of data is stored or processed on the mainframes; whether there is a separate test environment.

Emagined has been testing mobile applications since Apple first introduced mobile applications. We have experience testing both Android and iOS systems, and we have pentesters that specialize in each. The specific methods differ between both platforms. We have specific devices that have been set up specifically to allow us to work at the command-line level for these tests.

Some things we want to know are where the data are stored (on the device, on a server); how it is connecting to the web to connect data back and forth; whether or not the system that the mobile app is connecting to is also going to be tested; will we be able to get a copy of the application or download it from the Apple/Android store; do we need to put in an end-user's domain; do we have the third party vendor's permission to do the testing; the users in the environment so we can set up testing at those multiple levels; is there an ability to use non-production data when testing the mobile application; if the development staff are available; and what version of Android or iOS must the application run on.

There is always the chance of something going wrong during a penetration test. Hopefully it's something minor, like an incorrect URL or incorrect credentials. When we're doing scoping for our testing, we try to document things that can go wrong during the test so that we can avoid that scenario. Try to get application demos and all this content ahead of time. Around 4-5 years ago, we were testing some public exploit code for a Cisco device; we asked the customer for permission to test the exploit code, which we received. As a result of testing this exploit, we inadvertently cause the router to reboot, which took the network offline for about 15 minutes. From that event, one of the lessons we learned is that we express to all of our pentest leads that there has to be open and frequent communication with the customer. We should ask questions and validate information, and our testers should speak with their peers.

Emagined Security has 12 fulltime penetration testers, all but one of whom are CICS certified as of right now. All are US based, and Emagined Security is a CREST certified penetration testing firm. For a penetration test of this size, we could probably fit in 3 to 4 tests concurrently with this staff of 12 testers. One of the benefits to having lead time and good project management is that we could stagger the testing among different entities to move our staff around. Our goal would be to do the sample scope of work in about 2 weeks with a team of 3-4 penetration testers. Part of this short timeframe relies on having the logistics figured out beforehand, such as connectivity, credentials, etc. Our lead time is 2-3 weeks, but that assumes we can get everything coordinated in advance.

We find that a lot of cities and counties that we have done testing for have applications that are third-party owned and operated. We want to make sure that we have an introduction with a vendor in advance of planned penetration testing. We get authorization to test those systems through an ROE with that vendor or a letter of authorization from the vendor. We also aim to ensure that the customer agency/government has authorization to test an application (e.g., a Microsoft application), and that there aren't any requirements of that third-party vendor for penetration testing (e.g., what can and cannot be done). Some challenges we face are that there are times where we speak with the third-party, who then say they're okay with testing, but then become non-responsive. So we try to determine if there needs to be a different window of time for testing that third party application. In instances where we don't get permission from the vendor, what we instead do is give the customer a list of things they can ask from the vendor (e.g., prior pentests, ISO audits, FedRAMP certification, etc.) so they can gain some assurance over the application's security. There is also the report that needs to be communicated out; dissemination of the pentest results, including potential zero day vulnerabilities in a vendor-hosted system, would be at the discretion of SAO. Also key is clear and concise communication, including start/stop notifications, communicating high/critical vulnerabilities, daily status calls, etc.

Vendor: KPMG							
K689 - Security Assessment Services							
Interview Total 10 pts							
Question 1	Many of the state agencies and local governments we work with have thick client applications that we test. Can you describe your experience with penetration testing thick client applications? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a thick client?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 2	Our engagements can include testing of SCADA systems. In some cases this testing must be done in production on critical systems such as water or sewer treatment plants. Can you describe your penetration testing experience with SCADA systems? What are some of the key questions you would have for the client to ensure a smooth valuable test?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 3	Our engagements sometimes include testing mainframes. These would typically be legacy applications that support core or enterprise functions, and therefore may not be as stable or robust as more modern applications. Can you describe your experience with penetration testing mainframes? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a mainframe?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 4	Our engagements sometimes include mobile applications? Can you describe your experience with penetration testing mobile systems? What are some of the key questions you would have for the client to ensure a smooth and valuable test?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 5	Can you describe a time when something went wrong during a prior penetration testing engagement? How did you find out something went wrong? How did you respond or correct the situation? What lessons were learned from that event?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 6	From time to time we need to have more than one penetration testing engagement going simultaneously- with each engagement likely having internal and external testing with approximately 4-6 applications and network testing. Would your firm be able to staff this? If so, how much lead time would your firm require? How many penetration staff would you be able to commit to each engagement? How long would it take your staff to complete a typical penetration test of this size?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 7	Communication is an important part of penetration testing work, especially because the systems we test are not managed by SAO, but rather our auditees. Can you tell us about any experience you have testing a third-party application, for example a vendor for your direct client? What type of communication challenges did you face? How did you overcome those challenges?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Average Score	<div>6.86</div>						

Have extensive experience testing thick client applications, across a multitude of operating systems and code bases. Estimated we test about 50 of these per year. Typically what we look for in thick client applications are information disclosure vulnerabilities in file systems, and opportunities to manipulate the application to target the backend. The main goal of attacking a thick client is to use it to target the backend that it relies on, since the client by itself is limited in its functionality.

First thing we typically ask when testing thick client applications stems from the age/nature of these applications. So we ask what operating systems it can run on, what dependencies it requires to run, etc. Also ask what backend systems are in-scope (e.g., servers, APIs, etc.). Want to know who the points of contact are for that application (SMEs), so we can move forward in an informed way with the application. Also want to know if there are any credentials required to test the application.

Our experience in testing SCADA and OT systems is we do a lot of work at energy labs; been doing this for 3-4 years on-and-off; more of an ad hoc type test rather than an annual test, with the scope changing yearly. We do this for the DOE OIG. We have a team of sub-contractors that do this, as well as some staff who do this.

When it comes to testing SCADA or IoT, we need to know what kind of operating systems those systems are using so we know how to test them. Depends on what type of system it is, where does it sit, what does it control, how is it accessed, what kinds of users does it have, etc..

We've tested mainframes over the years for banking and defense functions, specifically to support large banking or payroll applications. A lot of the testing consists of shoulder surfing to ask administrators what they are doing. Since mainframes are often only in production, we need to be careful with them, and that's why we sit with the experts to make sure things are safe to test.

The key piece of information we would want to know is what is the security suite, so we can start to design our tests. This will sometimes come back to checking various STIGs and configurations, to check if those are appropriately configured. Can't really use many automated tools on mainframes (need to rely heavily on manual testing), which is why we rely on shoulder surfing with SMEs to make sure tests work well and safely.

We test both Android and iOS mobile applications. We test about 30 mobile applications per year. Our methodology is similar to our thick client methodology, in which we seek to escalate our privileges to affect backends and endpoints.

We ask if there are any backend APIs in-scope. If so, we'll ask for documentation on them so we can know how to approach it. We'll ask if there is both an Android and iOS component, as well as what version we're testing. Will also ask if there are any other dependencies (e.g., OS version, any jailbreak requirements, etc.).

We often have issues with penetration testing. We prefer to test in test environments, but sometimes testing in production is all that is possible. We had a recent example where a scan hit some legacy systems and caused a disruption, which we discovered once the affected system(s) stopped responding. We did try to ask about these beforehand, but this information was not provided to us. Once the outage happened, we were immediately in contact with the client to understand the scope of the problem. This allowed us to troubleshoot the issue quickly. Once we got the approval to reinstate testing, we made sure to exclude the network segment that experienced the outage.

Our team and program was built to scale. We can meet client needs immediately. We can handle two, four, six assessments at a time if needed. We have the appropriate staff within the U.S. to meet these timelines. We understand the expected lead time is 4-6 weeks, but can make 1-2 weeks work if needed. Number of staff per engagement will vary based on scope, as the test team might consist of specialized testers based on types of assets. For the sampled scope, we would estimate 4-6 weeks of testing.

We try to meet with the third party vendor at the earliest possible time, and we accomplish this by having a large number of staff available to meet with them early.

K689 - Security Assessment Services							
Vendor: Online		Interview Total 10 pts					
Question 1	Many of the state agencies and local governments we work with have thick client applications that we test. Can you describe your experience with penetration testing thick client applications? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a thick client?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 2	Our engagements can include testing of SCADA systems. In some cases this testing must be done in production on critical systems such as water or sewer treatment plants. Can you describe your penetration testing experience with SCADA systems? What are some of the key questions you would have for the client to ensure a smooth valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 3	Our engagements sometimes include testing mainframes. These would typically be legacy applications that support core or enterprise functions, and therefore may not be as stable or robust as more modern applications. Can you describe your experience with penetration testing mainframes? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a mainframe?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 4	Our engagements sometimes include mobile applications? Can you describe your experience with penetration testing mobile systems? What are some of the key questions you would have for the client to ensure a smooth and valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 5	Can you describe a time when something went wrong during a prior penetration testing engagement? How did you find out something went wrong? How did you respond or correct the situation? What lessons were learned from that event?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Question 6	From time to time we need to have more than one penetration testing engagement going simultaneously- with each engagement likely having internal and external testing with approximately 4-6 applications and network testing. Would your firm be able to staff this? If so, how much lead time would your firm require? How many penetration staff would you be able to commit to each engagement? How long would it take your staff to complete a typical penetration test of this size?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					6		
Question 7	Communication is an important part of penetration testing work, especially because the systems we test are not managed by SAO, but rather our auditees. Can you tell us about any experience you have testing a third-party application, for example a vendor for your direct client? What type of communication challenges did you face? How did you overcome those challenges?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
					8		
Average Score	7.71						

Described as extensive experience testing thick clients. Tested mostly applications built on Windows platforms. Initial things they need to know are what types of systems the client is hosted on (the vendor's machine, or the customer's machine); if installed on the customer's machine, need to know if they can install tools on that system so they can conduct the test. Try to do a walkthrough of the application from the client to see how it functions, what it accesses, to see if it reaches other APIs and web servers. And then test for different OWASP vulnerabilities, walking through the OWASP testing guide, and digging into the application itself (e.g., looking for hardcoded credentials in the application, looking through the actual code, etc.). Would also need credentials from the customer for the application.

For SCADA systems, we have done one water/sewer treatment plant before. Have also tested systems in the power industry at multiple power plants. When testing SCADA, we take a more tailored approach due to the fragility of the systems themselves. We work with the customer to identify which specific systems/IPs will be targeted, what those systems are, and what they do. We won't throw a full scan at those systems; instead, target specific ports and HMIs, look for default credentials. We use different tools for this purpose as well. Rules of Engagement will establish what times we can test to ensure a full team is available to respond in case anything does happen, though we do our best try avoiding disruption.

This would be approached similar to SCADA testing, establishing ROE and determining which systems are fragile in the environment. Trying to determine whether those systems **will** be tested or excluded from the scope altogether. When these systems are in scope, the exploits that would often be run against them can cause denial of service conditions. So it's important to establish a timeframe for attempting to run that exploit with explicit approval from the customer to confirm whether that system truly has that vulnerability. However, we have not really had mainframes listed out as a separate form of testing. They have typically been added in-scope or removed from the scope on prior engagements. We do have a number of clients that run mainframes, with varying levels of interfaces that connect to them. So our experience testing them is the nature of the different interfaces we're talking about. We also try to be very detailed in the ROE to outline the customer's responsibilities prior to testing.

We do have extensive experience testing mobile applications. We have a lot of clients that have built them or use them. We start by gathering information on whether it is an android application, iOS application, or both. We see if it's available through the app store, or if we need to retrieve the files themselves to conduct the testing. We would do a walkthrough beforehand to understand its functionality, and try to gather the API points to see how we can call those APIs. We try to gather the credentials we would need to access the application. Then we set the timeframe for the testing. We have a pretty mature process when doing the intake for testing; we have very specific spreadsheets that feed into defined Rules of Engagement that stipulate the conditions of testing (contacts, environment, etc.).

We've had some cases where we agree with a customer on something about being whitelisted through an IPS. Sometimes we have found that we haven't been whitelisted, and aren't discovering as many open ports as we would expect. We work with the client to try to see that the situation is rectified so that what we're seeing is expected. What we learned from that was to check with the client along the way, rather than keep forging ahead. This is something we would highlight during the daily progress reports as expected by this engagement. In another recent engagement, a customer gave us a scope that included some IDR systems that were well beyond their lifecycle. So when we went to test those systems, it caused some "problems." We talked with the customer about their infrastructure, determined they had multiple systems of a similar type, and we negotiated with them to take one of each type of system out of production for a late night period that we would test. We then tested those systems on a sample basis to get a better idea of what vulnerabilities existed across the broader population of those systems in the environment.

We have several customers that we work through on multiple applications (4-6) as well as network testing that we work on concurrently. So we will be able to meet that workload. What we try to do is have a balanced calendar, so we typically plan for 1-2 weeks out to ensure we have availability. The number of testers we would assign to each engagement would vary depending on the nature of the assets being tested (e.g., SCADA, mainframe, etc.). 6 applications plus network testing would be approximately two weeks, plus reporting time. We handle many pentests at the same time; just got done with a customer that had 4 applications (external and internal), with the engagement lasting three weeks.

We've had many customers that have wanted a third-party application tested because they know that having the third party application in the customer's environment, or hosted by that third-party, presents risk. So it comes down to communication about things like accounts, testing environment/data, etc. We've found sometimes there is a communications delay in terms of getting from the customer to the third-party. One way we've mitigated these delays is by setting up regular calls with the customer so that they can hear what we need from the third-party vendors. Making sure all three parties (the test team, the customer, and the third-party) are on the same page is important to avoid differing expectations. It's also helpful to walk the third-party through the pentest report so that the vendor can have a better understanding of things that need to be fixed urgently, versus less urgent findings.

K689 - Security Assessment Services							
Vendor: SAC							
Interview Total 10 pts							
Question 1	Many of the state agencies and local governments we work with have thick client applications that we test. Can you describe your experience with penetration testing thick client applications? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a thick client?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				4			
Question 2	Our engagements can include testing of SCADA systems. In some cases this testing must be done in production on critical systems such as water or sewer treatment plants. Can you describe your penetration testing experience with SCADA systems? What are some of the key questions you would have for the client to ensure a smooth valuable test?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
			2				
Question 3	Our engagements sometimes include testing mainframes. These would typically be legacy applications that support core or enterprise functions, and therefore may not be as stable or robust as more modern applications. Can you describe your experience with penetration testing mainframes? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a mainframe?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 4	Our engagements sometimes include mobile applications? Can you describe your experience with penetration testing mobile systems? What are some of the key questions you would have for the client to ensure a smooth and valuable test?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 5	Can you describe a time when something went wrong during a prior penetration testing engagement? How did you find out something went wrong? How did you respond or correct the situation? What lessons were learned from that event?						
	<div></div>						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
			2				
Question 6	From time to time we need to have more than one penetration testing engagement going simultaneously- with each engagement likely having internal and external testing with approximately 4-6 applications and network testing. Would your firm be able to staff this? If so, how much lead time would your firm require? How many penetration staff would you be able to commit to each engagement? How long would it take your staff to complete a typical penetration test of this size?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 7	Communication is an important part of penetration testing work, especially because the systems we test are not managed by SAO, but rather our auditees. Can you tell us about any experience you have testing a third-party application, for example a vendor for your direct client? What type of communication challenges did you face? How did you overcome those challenges?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Average Score	<div>4.57</div>						

For most thick clients when we test them, we do a couple different tests. We take a snapshot of the system that the software will be installed on, and then observe how that software affects the security of that system once installed and running. Also tend to check what kinds of privileges the software requires. Part of this is to see what is the least level of privilege that the application can run on. **(Note: The way this is being described is as a test of the host system (i.e., the Windows desktop), rather than a test of the application itself)**. Will also observe the security of the traffic as it is going between the client and the server. We've tested dozens in the past few years. A lot of the work we've done has been on thick clients that have tied into data warehouses. Have tested elections software as well. The response later mentioned some of the methods and techniques to test the **application** itself.

We test SCADA systems passively; connect a machine to a span port, and analyze traffic as it moves. This is for asset discovery. We can also use the analyzed traffic to get an accurate list of vulnerabilities on the network (e.g., software versions). A big part of OT and SCADA that we ask is how often the network is tested and updated; if tested/updated infrequently, we assume it to be more fragile than if it were tested and updated regularly. Another thing we ask is for a list of assets on the network, as well as information on it. A separate person in the interview said they need to do active testing and scanning (thus contradicting the prior respondent), so that testing can discover the failure point of those systems and test for denial of service conditions. (Note: The question states that testing is on production systems providing critical services, such as water/sewer treatment; **testing for denial of service conditions is NOT the intent of our testing, especially on SCADA systems**).

Respondent talked about methods rather than experience.

Respondent talked about methods, but did not specify much in terms of experience. Respondent also did not talk much about what specific questions they would ask, but instead spoke about mobile testing fitting within a general penetraton testing framework.

Respondent said something always goes wrong; it's a matter of the extent of the damage that something went wrong. You go in as a hacker and intend to break something, so things inevitably break. The key is to have compensating controls, and to make sure that you can back out of the process quickly. We try to monitor the targeted hosts as we conduct testing so that we can avoid breaking things. Respondent framed denial of service results as something they almost want to do to better understand the weaknesses in those systems to prevent further disruption. **(Note: Denial of service testing is not something our penetration testing aims to conduct, especially on systems that are tested in production)**. We do require an on-call person depending on the sensitivity of the system being tested.

Respondent said they are confident they have the capacity to meet this contract. They said they also recently completed a pentest of Pierce County, which included an extensive scope. Respondent did not specify a number of staff that they would be able to commit to each engagement, instead saying it would vary by size of scope. Same with the question regarding the time to complete a penetration test. Vendor is tapped into 3rd party associations and information sharing organizations (e.g., Water-ISAC), giving them insight into the tools, techniques, and approaches to take with varying types of organizations. We've had a fair bit of experience having to schedule into blocks of time so that we're not impacting the operations of our client, while still providing a valuable test. We do have limitations on being able to staff concurrent tests on SCADA environments, since our pool of qualified testers for those types of systems is smaller than our pool of network and web application penetration testers.

Submitting an information security plan well in advance, and having a go/no-go meeting prior to testing.

Vendor: Soteria							
Interview Total 10 pts							
Question 1	Many of the state agencies and local governments we work with have thick client applications that we test. Can you describe your experience with penetration testing thick client applications? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a thick client?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 2	Our engagements can include testing of SCADA systems. In some cases this testing must be done in production on critical systems such as water or sewer treatment plants. Can you describe your penetration testing experience with SCADA systems? What are some of the key questions you would have for the client to ensure a smooth valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				8			
Question 3	Our engagements sometimes include testing mainframes. These would typically be legacy applications that support core or enterprise functions, and therefore may not be as stable or robust as more modern applications. Can you describe your experience with penetration testing mainframes? What are some of the key pieces of information you would want from the client to ensure a smooth and valuable test of a mainframe?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				8			
Question 4	Our engagements sometimes include mobile applications? Can you describe your experience with penetration testing mobile systems? What are some of the key questions you would have for the client to ensure a smooth and valuable test?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 5	Can you describe a time when something went wrong during a prior penetration testing engagement? How did you find out something went wrong? How did you respond or correct the situation? What lessons were learned from that event?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				10			
Question 6	From time to time we need to have more than one penetration testing engagement going simultaneously- with each engagement likely having internal and external testing with approximately 4-6 applications and network testing. Would your firm be able to staff this? If so, how much lead time would your firm require? How many penetration staff would you be able to commit to each engagement? How long would it take your staff to complete a typical penetration test of this size?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				6			
Question 7	Communication is an important part of penetration testing work, especially because the systems we test are not managed by SAO, but rather our auditees. Can you tell us about any experience you have testing a third-party application, for example a vendor for your direct client? What type of communication challenges did you face? How did you overcome those challenges?						
Score	0-no answer	2-lack of understanding	4-Marginal	6-Average	8-Good	10-Excellent	Evaluator Score
				8			
Average Score	7.43						

Vendor has had several clients that have included thick client penetration testing. What they seek when they are preparing to test is define the rules of engagement to understand the full scope of the engagement; whether the thick client test includes the back-end, just the application itself, the traffic flows, etc.; want to understand the application by being provided with a copy of the application being assessed, as well as an overview of the application and its expected flow (a demonstration from a user's perspective). Then we'll set up a test environment that mimics the customer's environment so we can observe what it is doing to analyze the behavior of the system. We then explore the application and look for ways to provide inputs into the application (user inputs, uploaded files, etc.) as well as how data are transmitted.

We have a number of clients that operate these types of systems, as well as prior experience testing SCADA systems in water providers. When preparing for an engagement in that type of environment, while our overall approach is similar as penetration testing, we tread a lot more carefully because industrial systems react much differently. They're not always built with security in mind. The main thing we try to arrange as part of the engagement is communication. We need to fully understand the environment to the best of our abilities, getting walkthroughs of the topology of the environment because we don't want to accidentally interact with a system that they weren't aware of. The end result is a little different in this testing also because we stop short of exploitation on these tests; we stop at the point of showing that an attacker "could" attack a system, rather than outright demonstrating it.

We encounter mainframes from time to time, especially with state/local government and financial institutions, whether it's something explicitly identified as a component in the test or whether it's something we come across in the environment. We have a similar approach across different asset types, and can tailor it to mainframes. As with SCADA and OT, we need to be careful when testing mainframes. We need to have communication with the customer to become familiar with the system. We also converse amongst ourselves as a team to discuss potential approaches to test the mainframe if it was included in testing. If we stumble across it, we work with the customer to determine if it should be added to the scope. Exploitation of identified vulnerabilities occurs if we believe it can be done safely. We're looking for known vulnerabilities and access control issues.

When we're testing mobile applications, what we want to do first is understand the application, what it does, what the user workflow is, etc. Once we have done that, then we set up the test environment in order to have a controlled test environment where we can monitor the behavior of the application and network flows. We have mobile devices that are rooted specifically for testing. From there, it's a defined test where we look for places to inject data, interact with the application. For Android apps in particular, we also look to see if we can reverse engineer apps to find additiona vulnerabilities. One of our bigger state clients had mobile applications.

We work hard with our teams to make sure that they treat every potential problem seriously. During an nmap scan of a customer, we didn't think that our nmap scan caused the service disruption the customer reported. We worked with the customer and discovered a configuration error causing a routing loop in the firewall that was magnified by the nmap scan. We couldn't want for the customer to resolve the issue to complete the test, so we had to get creative and test the in-scope assets without traversing that misconfigured firewall. We have a practice of doing an AAR after any engagement that doesn't go smoothly, and this has informed many of our rules of engagement discussions.

Absolutely, would be able to staff this contract. Lead time will vary by size/scope of the engagement, number of other projects, etc. More lead time is obviously better, but we are very accustomed to having very quick turnarounds for projects. We also have no problems bringing in new staff to join our team if we see a continuing trend of increased engagement from our customers. We always have people applying and trying to join the team. If an engagement is less urgent, we may have one tester who collaborates with other team members for support; if it's more urgent, we might have more. 1-2 testers per engagement is typical for our engagements. Most of our engagements have 2-4 weeks of lead time, but that can vary. For the scenario given, we would typically allocate 3-5 weeks for testing, but we report as we test. So as we're going, we're building our report.

We've been contracted by elections commissions to test voting systems; the third-party vendor in these cases was often happy since they essentially received a free pentest of their system. But we work hard to ensure that we're not surprising vendors with the vulnerabilities/results of our testing, and to establish clear reporting protocols so that they know who will receive that information as well as how. We also sometimes have to work around the challenges of NDAs with vendors, since we work for the customer, not the vendor. When we're communicating with a client, if there's a critical finding, we're communicating that right away rather than waiting until the reporting timeframe.